ACDL Workshop on Graph Neural Networks
Sailab, University of Siena
22 July 2019

Neural-Symbolic Systems

Artur d'Avila Garcez

City, University of London

a.garcez@city.ac.uk

@AvilaGarcez

http://www.staff.city.ac.uk/~aag/

www.neural-symbolic.org

The search for richer neural structures

Trees, Graphs, Logics (relational, nonmonotonic, probabilistic, etc.)

Constraint-based Machine Learning (M. Gori, MIT Press, 2018)

Brain/Mind dichotomy

Symbolic AI: a symbol system has all that is needed for general intelligence

Sub-symbolic AI: intelligence emerges from the brain (neural networks)

Neural-Symbolic Systems

Cognitive Science

$\downarrow$

Logic
Learning
Neural Computation

$\uparrow$

Neuroscience

One Structure for Learning and Reasoning
In AI: KR+ML

Why Neurons and Symbols?

"We need a language for describing the alternative algorithms that a network of neurons may be implementing" L. Valiant

(New) Logic + Neural Computation

GOAL: Learning from experience and reasoning about what has been learned in an uncertain environment in a computationally efficient way.

Neural-Symbolic Methodology

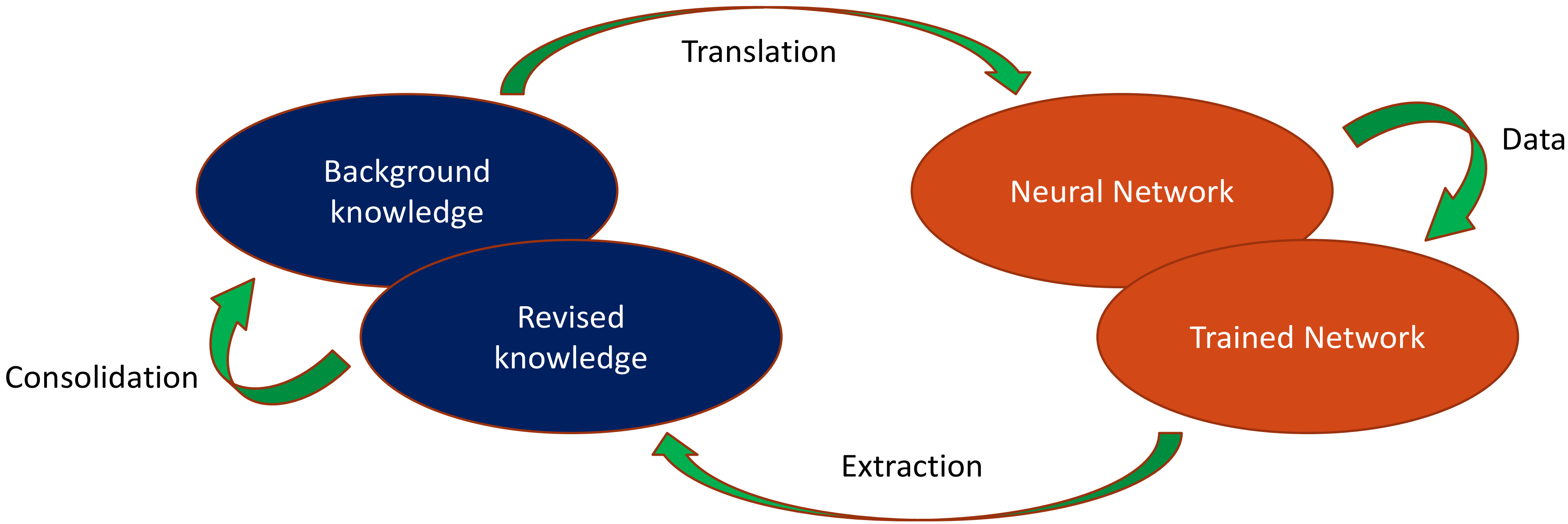
high-level symbolic representations
(abstraction, recursion, relations, modalities)

translations

low level, efficient neural structures
(with the same, simple architecture throughout)

Analogy: low-level implementation (machine code) of high-level representations (e.g. java, system requirements)

Neural-Symbolic Learning Cycle



Connectionist Inductive Logic Programming (CILP) System

A Neural-Symbolic System for Integrated Reasoning and Learning (neural nets + logic programming)

- Knowledge Insertion, Revision (Learning) and Extraction

  (based on Towell and Shavik, Knowledge-Based Artificial Neural Networks. AIJ 70:119-165, 1994)

CILP = backpropagation with background knowledge (BK)

- Applications: DNA Sequence Analysis, Power Systems Fault Diagnosis

  CILP test set performance is comparable to backprop.

  CILP test set performance on small training sets is comparable to KBANN and better than backprop.

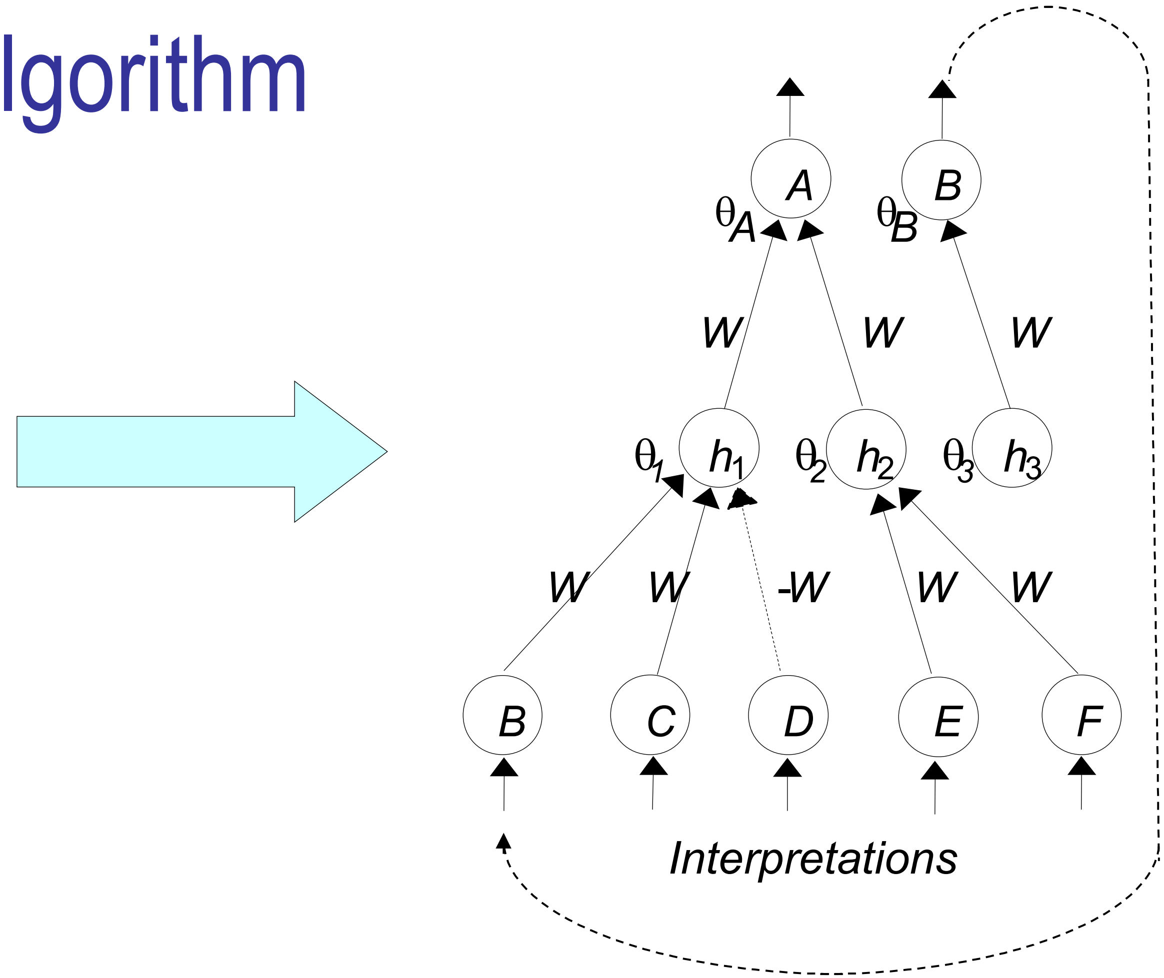
  CILP training set performance is better than backprop. and KBANN

CILP Translation Algorithm

r_1: $A \leftarrow B, C, \sim D;$

r_2 : $A \leftarrow E, F;$

r_3 : $B \leftarrow$



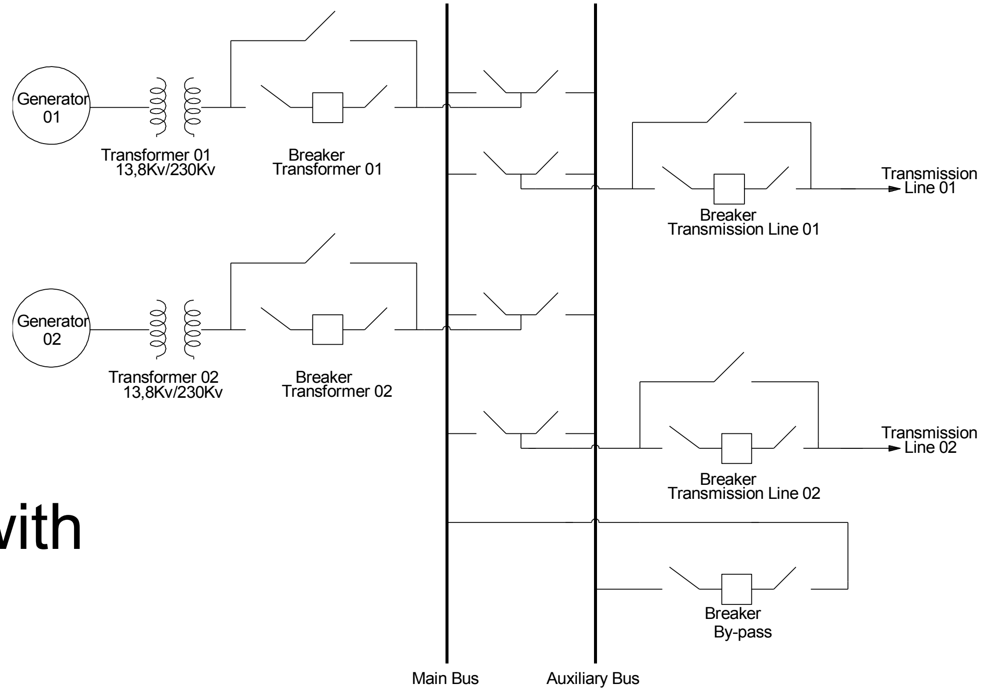
THEOREM: For any logic program P there exists a neural network N such that N computes P

based on Holldobler and Kalinke's translation, but extended to sigmoid neurons (backprop) and hetero-associative networks

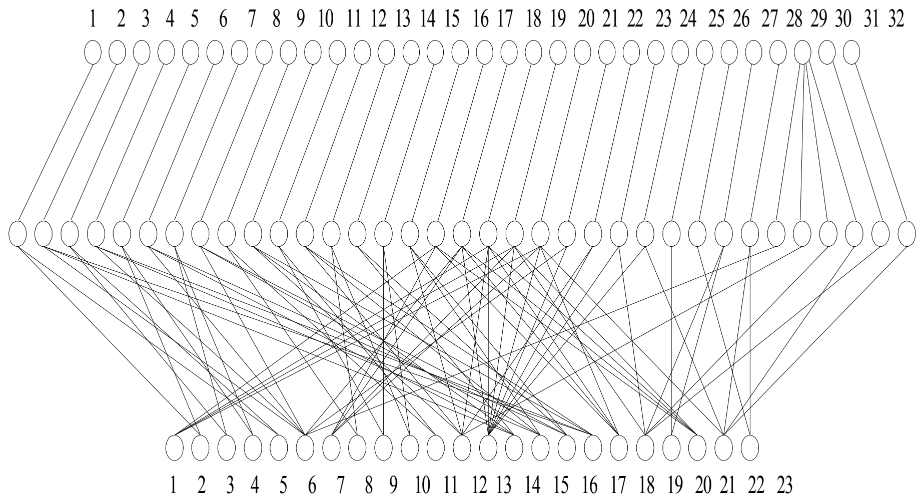
Holldobler and Kalinke, Towards a Massively Parallel Computational Model for Logic Programming. ECAI Workshop Combining Symbolic and Connectionist Processing , 1994.

Power Plant Fault Diagnosis

First real-world application of CILP



Mapping 23 alarms to 32 faults, with 35 rules (with errors) in the BK



Power Plant Fault Diagnosis

Background Knowledge (35 rules with errors)

278 examples of single and multiple faults

Fault(ground,close-up,line01,no-bypass) IF
  Alarm(instantaneous,line01) AND
  Alarm(ground,line01)

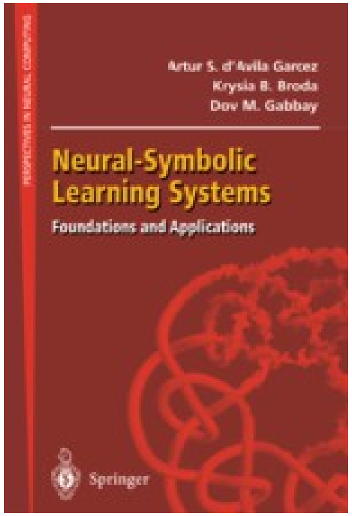
There is a fault at transmission line 01, close to the power plant generator, due to an over-current in the ground line of transmission line 01, which occurred when the system was not using the bypass circuit.

Power Plant Fault Diagnosis (results)

CILP achieves accuracy comparable to that of networks trained with backprop. or KBANN with the same BK, but it learns faster than both, and it performs better on smaller training sets (human-like computing?).

We attribute this to the soundness of the CILP translation (i.e. the above theorem; KBANN isn't provably sound).

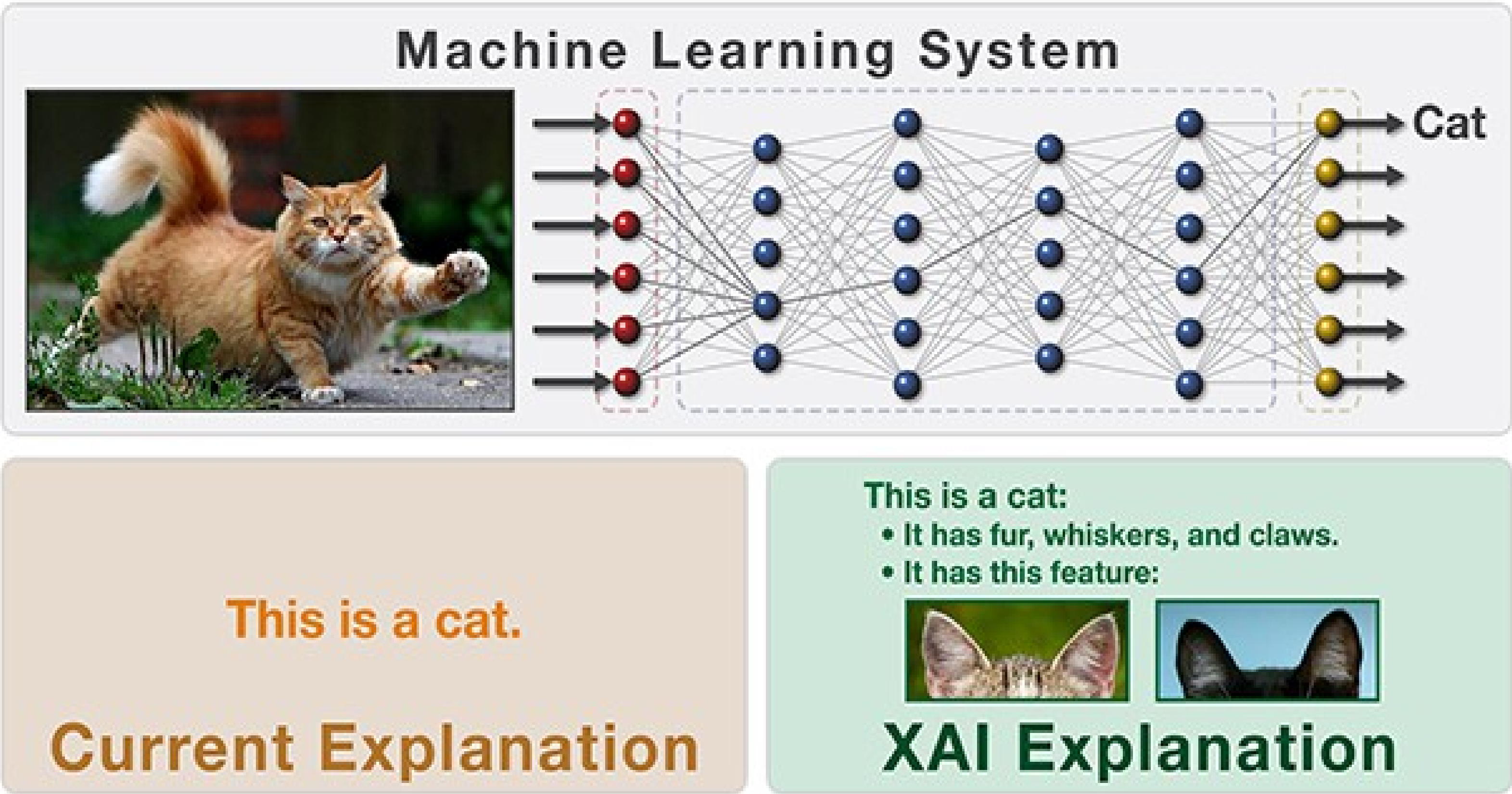
For details: Garcez, Broda and Gabbay, Neural-Symbolic Learning Systems, Springer, 2002.



The need for Knowledge Extraction

Proof history (goal-directed reasoning)
Levels of abstraction (modularity)
Transfer learning (analogy)
System maintenance/improvement

DARPA's Explainable AI



- XAI = Interpretable ML
- Explanation = knowledge extraction, not XAI

Knowledge Extraction techniques

- Soundness is important!
- Early methods: MofN, CILP
- Decision tree extraction - TREPAN
- Automata extraction - recurrent networks
- Current work: extraction from deep nets, soft decision trees, probabilistic MofN, distillation, local explanation methods... c.f. Human-like Computing 2019:

  http://mi21-hlc.doc.ic.ac.uk/programme.html

  Towards Providing Causal Explanations for the Predictions of a Deep Network

  http://mi21-hlc.doc.ic.ac.uk/short_presentations/White_Garcez.pdf

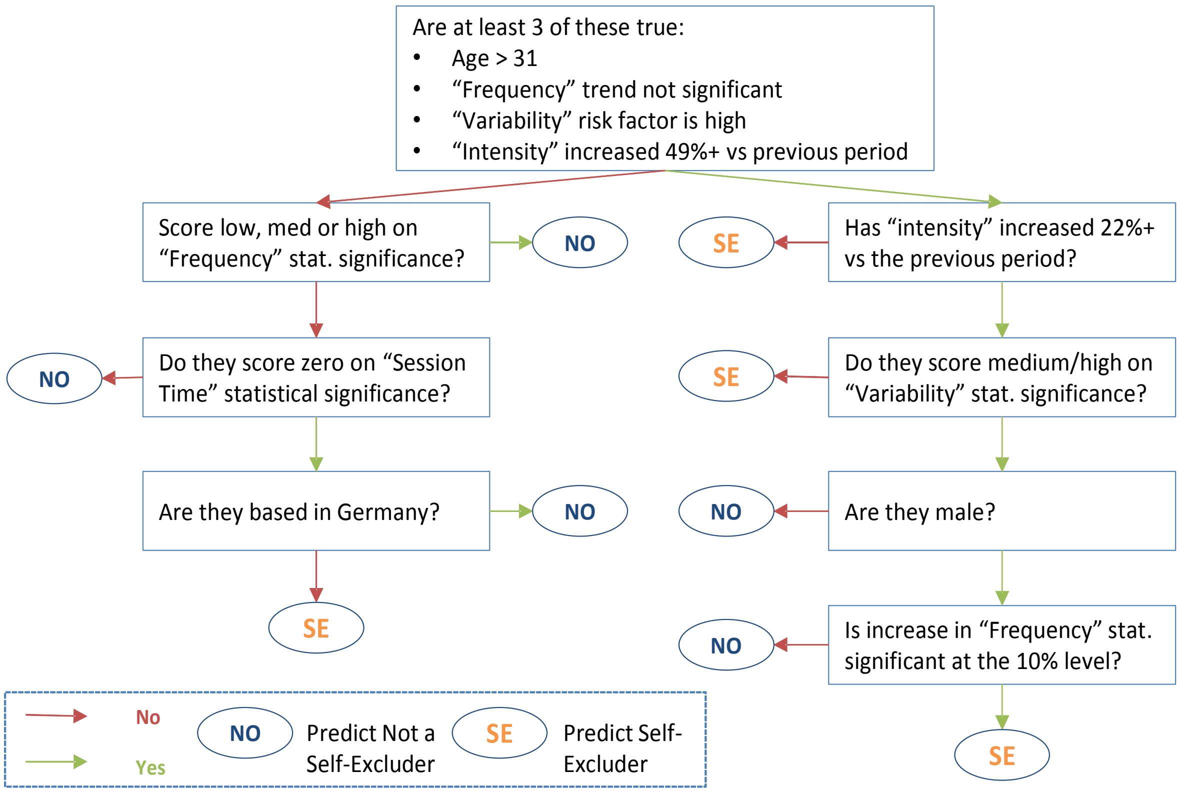
Reducing harm from gambling: a practical application of knowledge extraction

- 2014-16 EPSRC/InnovateUK project with BetBuddy ltd. (acquired by Playtech plc)
- Trained a neural net to predict whether someone should self-exclude from the game based on transaction data: frequency of play, betting intensity, variation, etc. (altogether some 25 markers)
- Used self-exclusion as a proxy for potential harm (avoids use of much more complex model of addiction)

Reducing harm from gambling

- Neural nets and Random Forests performed considerably better than logistic regression and Bayesian nets
- BetBuddy ltd. system is required to provide explanation to the regulator, gambling operator and to the player!
- Extracted decision tree can help debug the system and improve results too: "Are they based in Germany?"

TREPAN-style Knowledge Extraction:



C. Percy, A. S. d'Avila Garcez, S. Dragicevic, M. Franca, G. Slabaugh and T. Weyde. The Need for Knowledge Extraction: Understanding Harmful Gambling Behavior with Neural Networks, In Proc. ECAI 2016, The Hague, September 2016.

Ethical issues

- Recall our extracted decision tree: Are They Male? Yes/No; this is apparently illegal (c.f. UK car insurance case, gender cannot be a feature of the decision), although here as in medicine there is a good argument for using gender (AI4Good?)
- Much recent work on "which features to keep out so that ML system is ethical?"
- This is the wrong question... there are many unknown proxies in the data
- Our approach: make system interpretable instead and decide on whether or not to intervene! c.f. Rich Caruana's NIPS 2017-18 talks

It is not about neurons vs symbols...

- CILP is localist (and extends to nonmonotonic, modal, temporal, intuitionistic, epistemic logic, abduction, argumentation...)
- Alternative: distributed representation (e.g. Logic Tensor Networks (LTN); deals with first-order multi-valued logic)

- LTNs: Logic Tensor Networks: Deep Learning and Logical Reasoning from Data and Knowledge, Luciano Serafini and Artur d'Avila Garcez, Jun 2016 https://arxiv.org/abs/1606.04422

Logic Tensor Networks (LTNs)

- Neural nets with rich structure can represent more than classical propositional logic
- But neural nets are essentially propositional (i.e. do not use variables explicitly)
- To take advantage of full FOL, a more **hybrid** approach is needed
- One needs to get the representation right first: the logical statements act as (soft) **constraints** on the neural network learning…

LTN application: Semantic Image Interpretation

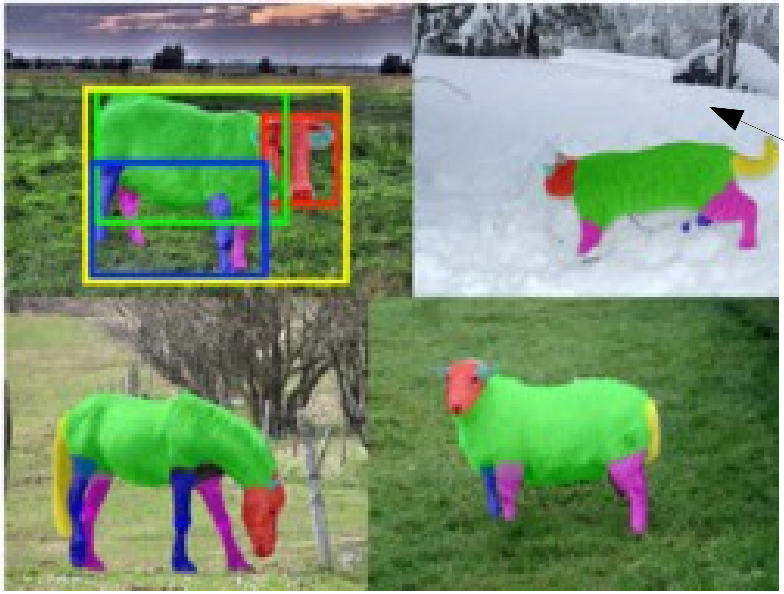
Given a picture extract a graph that describes its semantic content

Background knowledge:

Normally, every cat has a tail

Another example:

$$\forall xy(\mathsf{partOf}(x,y) \rightarrow \neg\mathsf{partOf}(y,x))$$



Make sure your system does not distinguish cats from wolves 99% correctly just because of the snow in the background...

Semantic Image Interpretation (Cont.)

In LTN, we build the graph by predicting facts given the bounding boxes, e.g.: Cow(b1), PartOf(b2,b1), Head(b2), etc.

In LTN, an object is described by a vector of features: e.g. John = (NI number, age, height, 3x4 picture, etc.)

Object detection (bounding box detection and labeling) is performed by an object detector (Fast RCNN)

LTN assigns a degree of truth (the grounding G) to atomic formulas: G(Cow(b1)) = 0.65, G(PartOf(b2,b1)) = 0.79...

G(bi) = <score(Cow), score(Leg) … score(Head), x, y, x’, y’>

Semantic features: the score of the bounding box detector on b_i for each class of objects

Geometric features: the coordinates of b_i

LTN in action

1. $\forall x(\neg PartOf(x,x))$
2. $\forall xy(PartOf(x,y) \rightarrow \neg PartOf(y,x))$
3. $\forall xy(Cow(x) \wedge PartOf(x,y) \rightarrow Leg(y) \vee Neck(y) \vee Torso(y) \vee Head(y))$
4. $\forall xy(Cow(x) \rightarrow \neg PartOf(x,y))$
5. $\forall xy(Torso(x) \rightarrow \neg PartOf(y,x))$.

- Grounding for PartOf is given by the % of intersection between two bounding boxes
- One can query the knowledge-base (KB) to obtain further groundings for training
- Learning is... maximizing satisfiability!

Learning in LTNs...

Given a KB and groundings, LTN calculates a grounding for the entire KB compositionally in the "usual ways"...

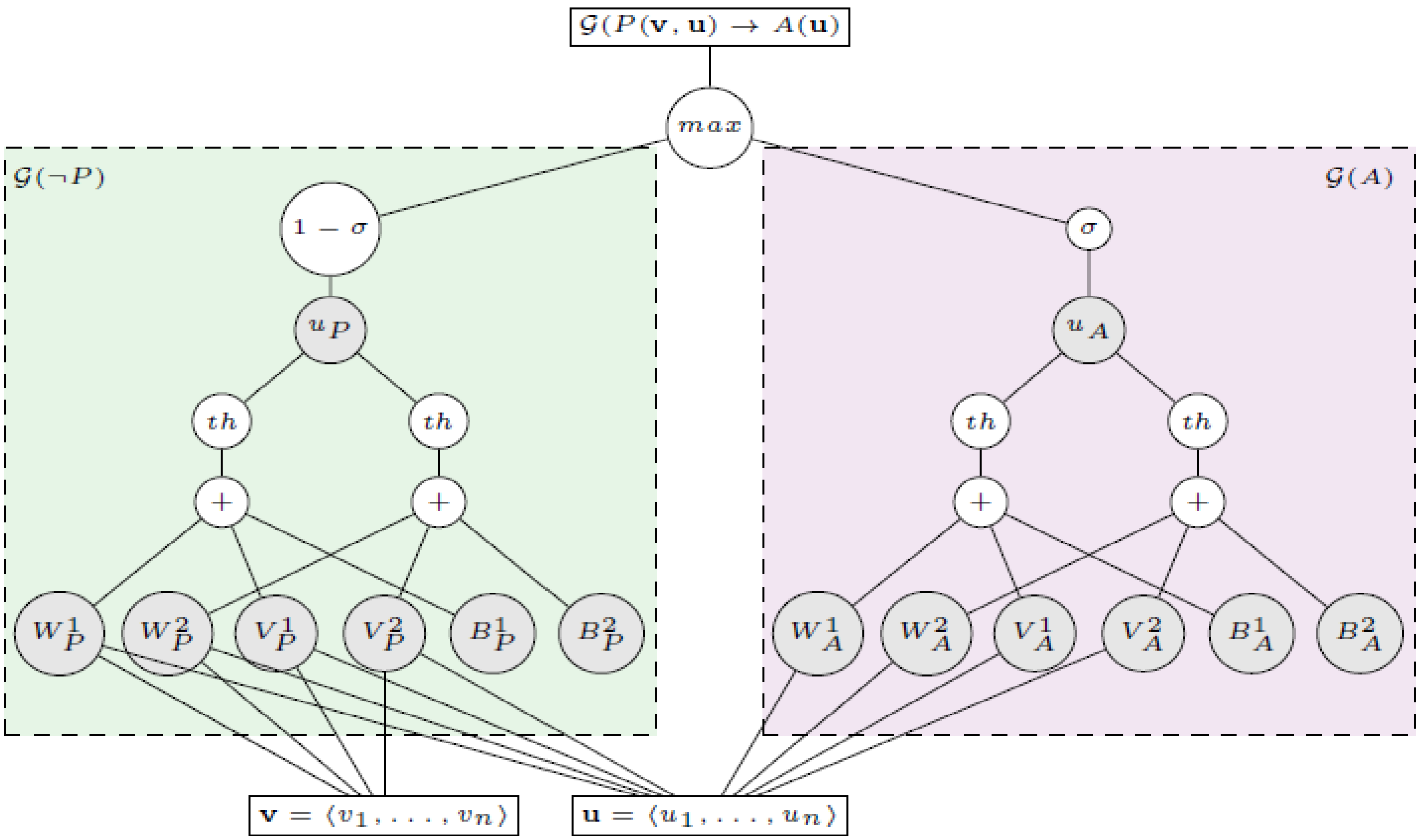


Fig. 1. Tensor net for $P(x, y) \rightarrow A(y)$, with $\mathcal{G}(x) = \mathbf{v}$ and $\mathcal{G}(y) = \mathbf{u}$ and $k = 2$.

The Tensor Network...

$$\mathcal{G}(f)(\mathbf{v}_1, \ldots, \mathbf{v}_m) = M_f \mathbf{v} + N_f$$

$$\mathcal{G}(P) = \sigma\left(u_P^T \tanh\left(\mathbf{v}^T W_P^{[1:k]} \mathbf{v} + V_P \mathbf{v} + B_P\right)\right)$$

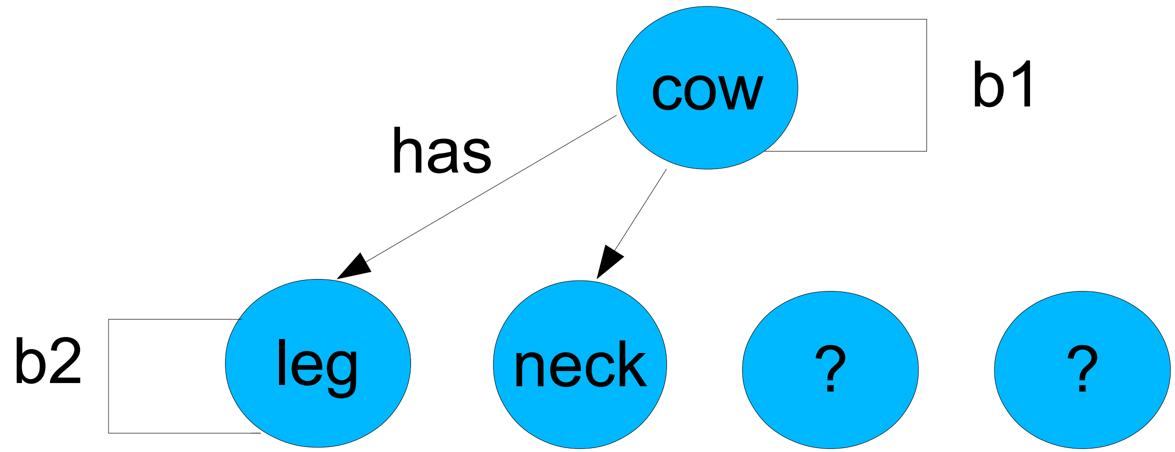
$$\mathcal{G}^* = \underset{\hat{\mathcal{G}} \subseteq \mathcal{G} \in \mathbb{G}}{\operatorname{argmin}} \sum_{\langle [v,w], \phi(\mathbf{t}) \rangle \in \mathcal{K}_0} Loss(\mathcal{G}, \langle [v,w], \phi(\mathbf{t}) \rangle)$$

Fast RCNN + LTN improves on Fast RCNN (state of the art at the time) at object type classification:

I. Donadello, L. Serafini and A. S. d'Avila Garcez. Logic Tensor Networks for Semantic Image Interpretation. In Proc. IJCAI'17, Melbourne, Australia, Aug 2017.

And finally, the knowledge graph...

- Given a trained LTN, start with an unlabeled graph.
- For every bounding box bi ask the LTN for the set of facts {Cow(bi), Leg(bi), Neck(bi), Torso(bi),...} and select the facts with grounding larger than a threshold.
- For every bounding box bi ask the LTN for the set of facts {PartOf(bi, bj)} with j = 1,…,n. Then, select the facts with grounding larger than a threshold.



Related Work

Compare and contrast with:

- Markov Logic Nets (MLNs)
- Bayesian Logic (BLOG)
- Inductive Logic Programming ILP-based approaches e.g. Deep ProbLog, dILP
- Probabilistic Programming (WebPPL)
- Lifted statistical relational AI e.g. Sum-Product Nets, Probabilistic Soft Logic, etc...
- Meta-interpretive Learning (higher order logic)

LTN is general

- Can be used for unsupervised learning, classification, regression, knowledge completion, learning embeddings…
- For more information c.f. EurAI ACAI 2018 Summer School on Statistical Relational AI, Aug 2018, Ferrara, Italy
- https://www.youtube.com/watch?v=RFFC9GD0po0&list=PLJPXEH0boeNDWTNwWTWnVffXi5XwAj1mb&index=15

Recent developments

On the Capabilities of Logic Tensor Networks for Deductive Reasoning. Federico Bianchi and Pascal Hitzler, AAAI Spring Symposium 2019, Stanford University, March 2019.

http://ceur-ws.org/Vol-2350/paper22.pdf

Injecting Prior Knowledge for Transfer Learning into Reinforcement Learning Algorithms using Logic Tensor Networks, Michael Spranger, IJCAI 2019 Workshop on Neural-Symbolic Learning and Reasoning, NeSy19, Macau, Aug 2019.

https://sites.google.com/view/nesy2019/home

Neural-Symbolic Computing

- Neural networks provide the machinery for effective learning and computation
- Perception alone is insufficient: AI needs reasoning, explanation and transfer
- Rich knowledge representation models: nonmonotonic, relational (with variables), recursion, time, uncertainty…
- Neural-symbolic computing: neural networks with logical structure (**compositionality**)

Conclusion: Why Neurons and Symbols

To study the statistical nature of learning and the logical nature of reasoning.

To provide a unifying foundation for robust learning and efficient reasoning.

To develop effective computational systems for integrated reasoning and learning.

Thank you!